REMARKS

This is intended as a full and complete response to the Office Action dated April 5, 2005, having a shortened statutory period for response set to expire on July 5, 2005. Reconsideration and allowance of the claims pending in the application is requested for reasons discussed below.

In this office action, the specification was objected to because the claims were not the object of a sentence starting with "the invention claimed is" or the equivalent. Therefore, correction has been made.

Further, claims 6 and 16 were rejected as being indefinite; therefore, these claims have been reviewed and revised as necessary to eliminate this rejection.

All of the claims 1-22 in the application were rejected under 35 U.S.C. 103 as unpatentable over various combinations of *Duluk 1* (U.S. 6,288,730) in view of *Duluk 2* (U.S. 6,771,264 B1) and considered further with *Baldwin* (U.S. 5,815,166 A) and *Swanson* (U.S. 5,421,028 A). These rejections are respectfully traversed.

To cover the many different aspects of the invention, applicant has cancelled claims 6-10 and 17-22 and submits new claims 23-32. Consideration and allowance of these claims is requested.

Duluk '730 describes a graphics processor in which position conflicts are alleged to be detected. However, the Examiner concedes that the program instructions described in the reference are only for reordering memory addresses being accessed and that there is no teaching that the program instructions are for configuring a fragment processor or delaying a fragment process. The Examiner attempts to supply this deficiency by citing *Duluk* '264. This reference describes Open GL which includes per fragment operations that define how pixels are stored. However, this reference also teaches nothing about reordering operations to take account of conflict detection. The combination does not teach or suggest eliminating flush instructions or re-ordering execution of fragment process instructions.

The Examiner further cites *Baldwin* '166, but only for a teaching of a fragment processing pipeline in which data is stored in an address specified by a pixel load

instruction. This reference also does not teach anything about resolving conflict issues in a manner relevant to the present application.

Finally, the Examiner cites *Swanson* '028 for *Swanson's* discussion of pipeline latency which, according to the Examiner, determines the duration of the pipeline flush. As acknowledged by the Examiner, *Swanson's* response to recognizing this problem is to re-synchronize his system, obviously a far more complex and time-consuming strategy.

Conventionally, a shading pipeline, such as a fragment processing pipeline included in the graphics processor, is flushed (using a flush instruction) prior to executing fragment program instructions to avoid any read or write (RAW) conflicts. A RAW conflict exists when a write to a position (for example, within a frame buffer or the like) is pending when a read from the same position is ordered. The RAW conflict is removed when the write to the position within the buffer or the like is completed.

In various embodiments disclosed and claimed herein, a flush is not needed between writing a buffer and reading a buffer. The necessity of the flush is minimized or deleted by performing a conflict check on the position, pixel, or other object of the write while before or while a fragment is processed. If the conflict check indicates that a conflict either exists or is possible, then the processing of that particular fragment is delayed. However, rather than executing a flush, as has previously been done, in this invention, other fragments continue to be executed while the processing of the fragment is delayed. When it is determined that the conflict has been eliminated, the processing of the delayed fragment is completed. In this way, the necessity of constantly providing flush instruction is eliminated. Such an approach is not taught or suggested by the references cited. As already recognized in the Examiner's office action, only one of the references, *Swanson*, even contemplates the problem of latency in execution of fragment processing; and even *Swanson* adopts an entirely different approach than that disclosed and changed herein.

In view of these clean distinctions, reconsideration and allowance of the claims is requested.

Having addressed all issues set out in the office action, Applicants respectfully submit that the claims are in condition for allowance and respectfully request that the claims be allowed.

Respectfully submitted,

James A. Sheridan
Registration No. 25,435
MOSER, PATTERSON & SHERIDAN, LLP
3040 Post Oak Boulevard, Suite 1500
Houston, TX 77056-6582
Telephone: 713-623-4844
Facsimile: 713-623-4846
Attorney for Applicants